Açai: a backup protocol for Lightning Network wallets

Scaling Bitcoin 2019 "Yesod"
September 11th-12th Tel Aviv

Margherita Favaretto
M.Sc.Eng. in Computer Science and Engineering

DTU Compute
Department of Applied Mathematics and Computer Science

Presentation Outline

1. Problem Description
2. Related Work
3. Preliminary concepts
4. Methodology
5. Açai Protocol: Design and Implementation
6. Conclusion and Future Work

Problem Description

Problem Description

The Lightning Protocol does not offer a decentralized, trustless recovery mechanism of bitcoins in case of wallet failure.

Problem Description- Scenario 1

Scenario 1:

"I accidentally deleted my lightning app and lost my channels. How can I safely recover my funds?" - Alice

Problem Description- Scenario 2

Scenario 2:

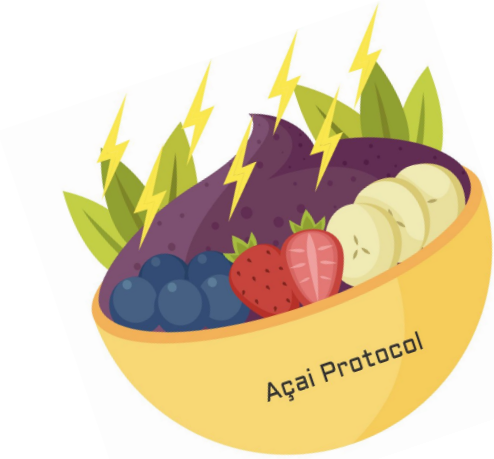
"I'd like to move my Lightning app to another machine. How can I safely recover my funds?" - Bob

Problem

The absence of BIP 39 and BIP 32 in Lightning Network makes impossible the trustless recovery of unspent transactions.

No portability among different devices and wallets.
No possibility to recover funds inside the Lightning Network if the device is damaged or lost.

Related Work



Related Works

Third-party Backup Mechanism	Self Backup Mechanism	Eltoo?
Ex. - Lightning wallet - Electrum wallet - Olympus	Ex. - Piln - Breez - Static backup channel	

Third-party Backup

Main idea:

- A third-party cloud service stores the information related to the channels

Weaknesses:

- Censorship Risk and Availability
- Centralization
- On cloud security threats
- Privacy

Technologies: Lightning wallet, Electrum wallet, Olympus

Self Backup

Main idea:

- The user is responsible of his own backup solution

Weaknesses:

- User experience
- Cloud Services Privacy
- Data loss risk

Technologies: Piln, Breez, Static backup channel

Eltoo?

<u>Main idea</u>:

The two sides (e.g. *Alice* and *Bob*) of a channel share the same commitment.

<u>Weaknesses</u>:

Bob might not be cooperative, sending to Alice a previous channel state to trigger the penalty.

Can we do better?

1. ***Decentralized*** system
2. ***Anonymity***, ***integrity*** and ***confidentiality***
3. ***Simple*** implementation
4. ***Censorship Resistant*** recovery service

Preliminary concepts

Açai Protocol

Main Goal ➡ Minor modifications to available protocols

Açai Protocol is based on the following concepts:

- Eltoo (*seen in* *Related Works*)
- Watchtowers
- BIP 32, BIP 44, BIP 39

Recovery Mechanism for Bitcoin Wallet

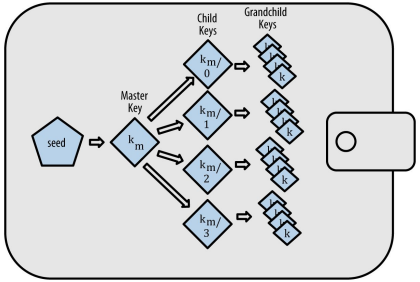
Wallets: Data structure used to store and manage a user's keys.

Deterministic wallet:

- All the keys are derived from a single master key, known as the **seed.**

HD wallet:

- BIP-32 standard for the public key calculation
- The most used key derivation



BIP-44:

- Standard to define a specific logical hierarchy for the HD wallet
- m / purpose' / coin_type' / account' / change / address_index

BIP-39:

Seed in hex: *0C1E24E5917779D297E14D45F14E1A1A*

Seed as a sequence of the following words (easy to remember and note on paper):

army van defence carry jealous true garbage claim echo media make crunch

Eltoo

Main idea:

The two sides (e.g. *Alice* and *Bob*) of a channel share the same commitment.

Weaknesses:

Bob might not be cooperative, sending to Alice a previous channel state to trigger the penalty.

Watchtowers

Use of ***Watchtowers*** as a mechanism of back up.

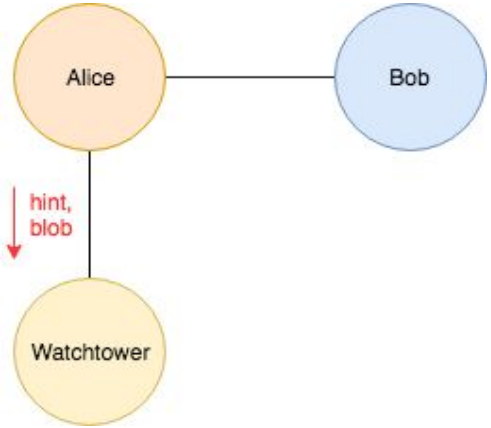
Watchtowers ➡

- Full nodes
- Always online
- Monitor status channel when a node is offline
- They will store eltoo channels

Watchtowers for monitoring status channels

Alice sends to the Watchtower WO the current status channel

1. Channel Status changes.
2. Calculate **txid** as the hash of commitment with Bob.
3. Calculate **hint= txid[:16]**.
4. Calculate **blob= Enc(data,txid[16:])**.
5. Alice sends to the Watchtower W0.

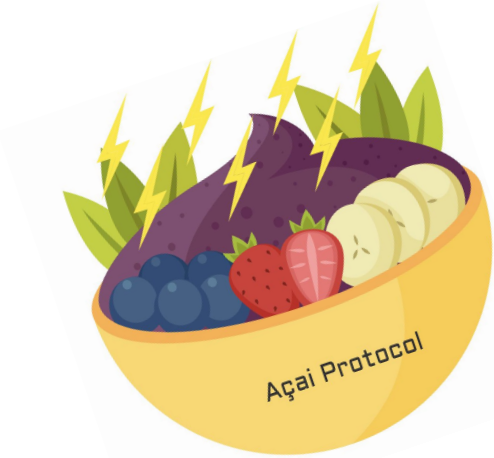


Watchtowers for monitoring status channels

Bob broadcast an older status channel

1. Bob broadcasts a transaction containing an older channel state.
2. The Watchtower examines the commitment broadcast by Bob.
3. The Watchtower notices that the txid[:16] equals to one of the past hints.
4. The Watchtower decrypts data, using txid[16:].
5. The Watchtower broadcasts to the Blockchain the justice transaction.

Methodology



Methodology

Açai Protocol Design →

- Game Theory and Adversarial Thinking
- Bitcoin Cryptography Standards
- Lightning Network Community

Methodology

Game Theory and Adversarial Thinking

- "*Formalizing and Securing Relationships on Public Networks*" by Nick Szabo (1997)
- "*A Cypherpunk's Manifesto*" by Eric Hughes (March 9, 1993)

Source: DTU Electronic Library/IEEE Xplore/ Google Scholar/Nakamoto Institute

Methodology

Cryptography

- "*Bitcoin: A Peer-to-Peer Electronic Cash System*" by Satoshi Nakamoto (October 31, 2008)
- "*The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments*" by Joseph Poon Thaddeus Dryja (January 14, 2016)

Source: DTU Electronic Library/IEEE Xplore/ Google Scholar/Nakamoto Institute

Methodology

Lightning Network Community

[Lightning-dev] Açai: a backup protocol for Li

Margherita Favaretto favarett.margherita at gmail.com
Sun Nov 18 03:13:43 UTC 2018

- Previous message: [Lightning-dev] RBF and dual-fund interactions
- Next message: [Lightning-dev] Rendez-vous proposal with ephemeral key switch
- **Messages sorted by:** [date] [thread] [subject] [author]

Hello, lightning dev community,

I'm writing to you to share an update of my Master Thesis project (previous
ightning network 11/12/2018),
positive in the Lightning network.

omknecht and Alex Bosworth for the
for the feedback to my previous
portant to proceed with my work.

fy me of the problem with my
:-)

the recovery of the unspent
rk after a wallet failure (e.g.
to the wallet storage).

eds and confidentiality for
the underlying distributed ledger
transactions through the
P32 address derivation.

use the watchtowers not just for
ckup service in order to solve the

previous e-mail, I've abandoned
lata in the watchtower, and I've

For simplicity
https://github
find all the d
add your comments and feedback.
<https://github.com/margheritafav/LightningNetworkProject>

Why double spend attacks on Lightning are not possible

Thursday Oct 25, 2018 by Ponvang Bulus
LIGHTNING SECURITY

Margherita Favaretto, a student working on remediation protocol for Lightning Network double-spend attacks asked for feedback for a proposed solution to double spend attacks using a "trusted remediation" gossip protocol.

ZmnSCPxj pointed out that double spend attacks are not possible on the Lightning Network unless both parties involved in the channel agree to it, which is not likely, first because the man at the other end of the channel will lose money. Secondly even if the other end of the channel is irrational enough to help the other guy double spend, they will still ask for an invoice and give the money using "existing invoice-payment mechanisms." ZmnSCPxj added:

> If the problem you are trying to solve, is the inadvertent publication of revoked commitment transactions, then the correct solution is not to have revocable transactions in the first place, i.e. eltoo. While it can be argued that it would take time for needed features of eltoo to appear on the blockchain layer (SIGHASH_NOINPUT_UNSAFE), it would also take time to implement "trusted remediation", by which time the problem could be solved

⚡ **Bitcoin Lightning Network Hackday New York City** ⚡

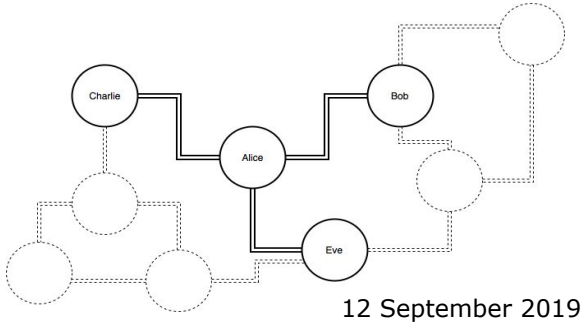
Açai Protocol

Açai protocol

<u>Idea</u>: Use the watchtowers not just for monitoring the channels, but also as a backup service

Standard Format : **hint= txid[:16]**
blob= Enc(data, txid[16:])
where txid is the commitment hash

Açai Format : **hintç= txidç[:16]**
blobç= Enc(dataç, txidç[16:])
where dataç=[txid_Bob, txid_Charlie, txid_Eve]



Txidç Derivation

Standard Format: *txid* is the hash of the commitment.

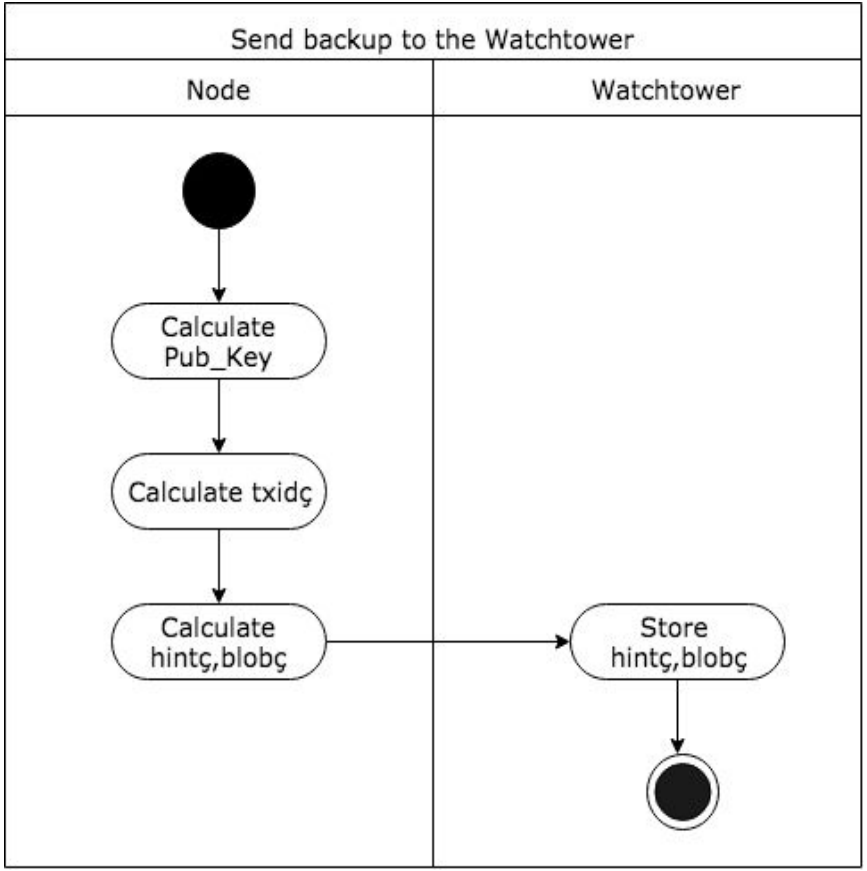
Value for *txidç*?

- Adopt BIP 39, BIP 32 using in Bitcoin
- BIP 44 (m / purpose' / coin_type' / account' / change / address_index)

Derivation Path for Açai Protocol:
m’/108’/0/(account_number)’/0/Current_Blockheight

txidç = 2SHA256(pub-key)

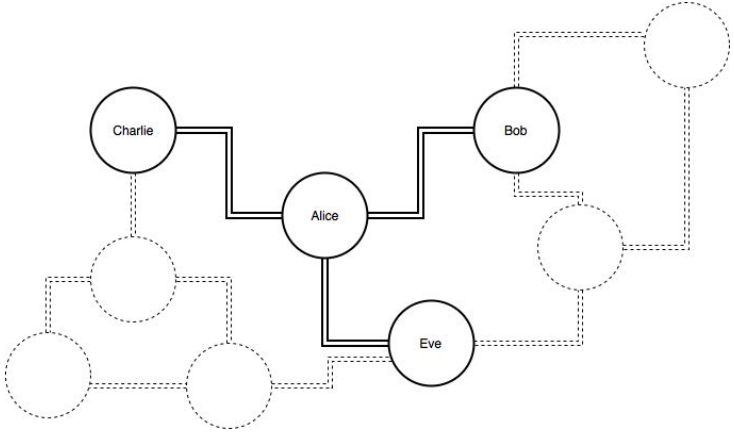
Açai protocol: how to send data



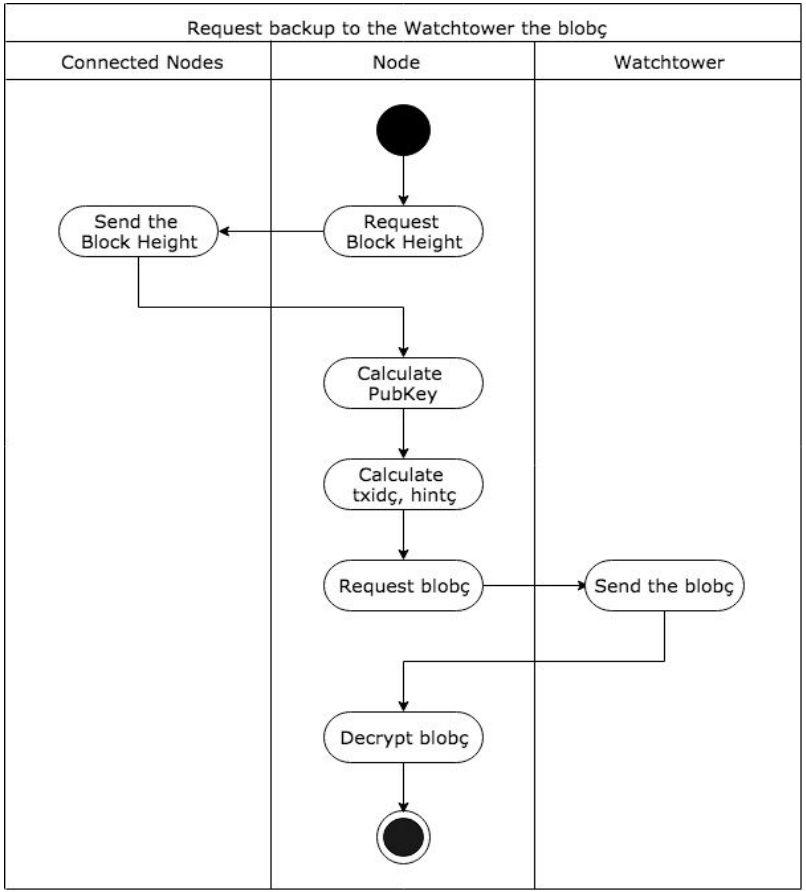
After each Channel Status change.

Derivation Path = m'/108'/0/(acc.number)'/0/Current_Blockheight

txidç= 2SHA256(pub-key)
hintç= txidç[:16]
blobç= Enc(dataç, txidç[16:])
dataç=[txid_Bob, txid_Charlie, txid_Eve]



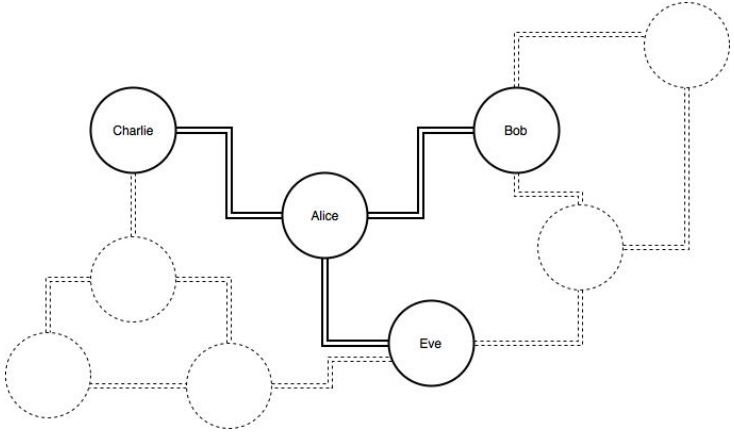
Açai protocol: how to recover data



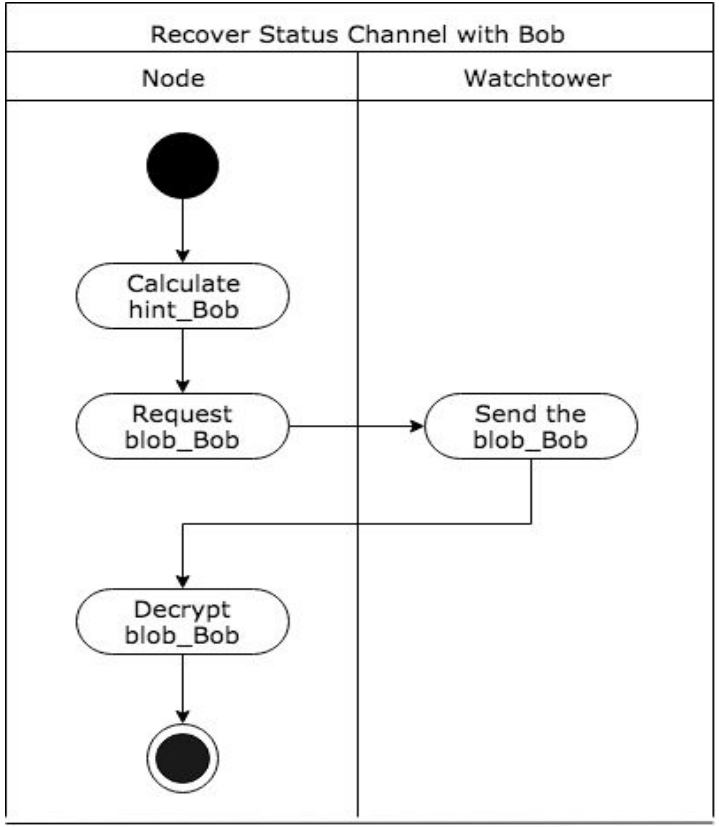
Note: Simplified version of the Activity Diagram.

Derivation Path = m'/108'/0/(acc.number)'/0/Current_Blockheight

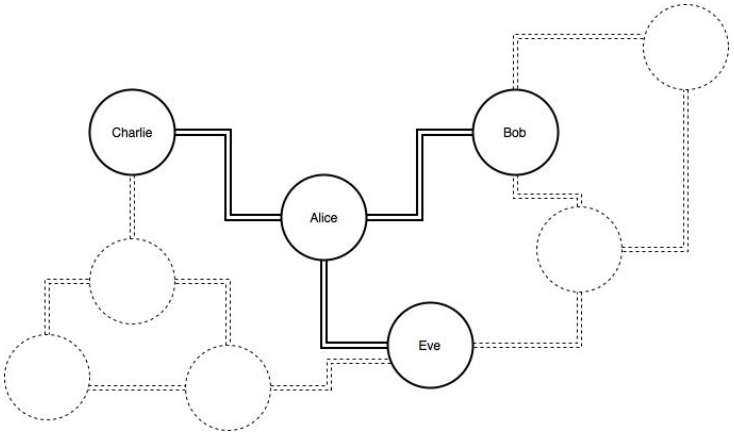
txidç= 2SHA256(pub-key)
hintç= txidç[:16]
blobç= Enc(dataç, txidç[16:])
dataç=[txid_Bob, txid_Charlie, txid_Eve]



Açai protocol: how to recover data



txid_Bob found through the Açai Protocol
hint_Bob= txid_Bob[:16]
blob_Bob= Enc(data_Bob, txid_Bob[16:])
data_Bob back up of the channel status with Bob



Açai protocol: assumptions

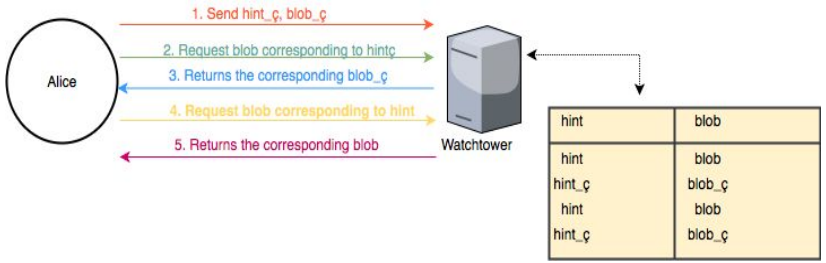
- Query function for the watchtower: there's a functionality enabling Lightning wallets to exchange data with a watchtower.

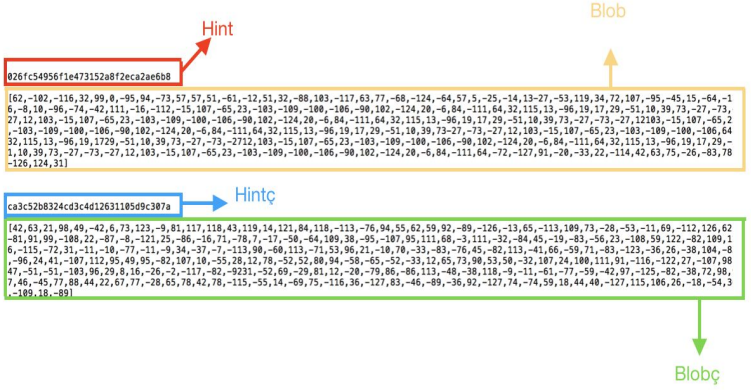
- Fees: Watchtowers benefit from trustless payments to become largely available, e.g. to cover the extra storage and bandwidth for the service (both for the normal service and the Açai backup).

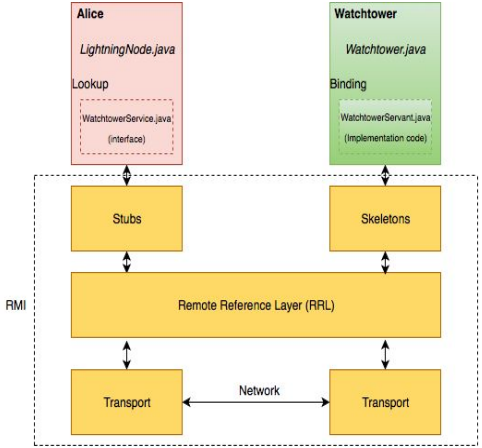
- Storage: Açai blobs stored in the watchtowers are not deleted, replaced/tampered.

- Directory: Alice knows which Watchtowers can provide her Açai backups.

Implementation



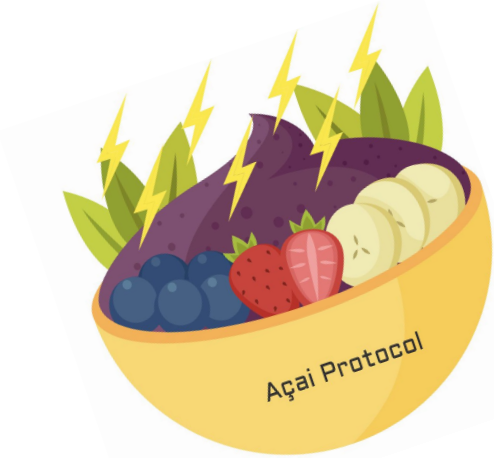




Results:

- BIP32 and BIP 39: Backup both for Bitcoin wallet and for Lightning wallets
- Portability between Lightning Wallets

Conclusion and Future Works



Results

- The Açai Protocol allows having a mechanism to backup data for a Lightning Network wallet.
- It satisfies all the proposed goals:

    1. ***Decentralized*** system
    2. ***Anonymity***, ***integrity*** and ***confidentiality***
    3. ***Simple*** implementation
    4. ***Censorship Resistant*** recovery service

- The Lightning nodes are able to recover all funds both in their Bitcoin Wallet and in their Lightning Wallet, through their own mnemonic seed (***bip39***).
- Formalization of Watchtower definition.

Definition of Watchtower

Formalization of Watchtower

We define Watchtower as a full-node, always online, that watch for Lightning channel breaches even at times when your wallet is offline. Watchtowers, by leveraging the Açai Protocol, provide a backup service if your Lightning-enabled Bitcoin Wallet must be recovered.

Açai:
a Protocol for frequent or sporadic users?

Problem:

For normal users is high consumption calculate the txidç using the value of the Block height in the Derivation Path.

Derivation Path= m’/108’/0(mainnet)/(account_number)’/0/Current_Blockheight

Solution:

Use as the last parameter a counter.

Derivation Path= m’/108’/0(mainnet)/(account_number)’/0/counter

where counter=0,1,2...n

Future Works

- Stack Sats enabling Açai on your Watchtower
- Lightning Wallet including the Açai Protocol
- Introduce Watchtower + IPFS to optimize the Açai Protocol
- Trade-off solution for frequent and sporadic users

Thank you! Questions?

Contact:
Margherita Favaretto
(fav.margherita@gmail.com)